Fundamentals of Programming in Visual Basic (Continue VI)

- **String Properties and Methods:**

  "Visual".Length is 6.

  "Visual".ToUpper is VISUAL.

  "123 Hike".Length is 8.

  "123 Hike".ToLower is 123 hike.

  "a" & " bcd ".Trim & "efg" is abcdefg.

Positions in a String

Positions of characters in a string are numbered 0, 1, 2, ….

Consider the string "Visual Basic".

Position 0: V

Position 1: i

Position 7: B

Substring "al" begins at position 4

Substring Method

Let str be a string.

str.Substring(*m, n*) is the substring of length *n*, beginning at position *m* in *str*.

"Visual Basic".Substring(2, 3) is "sua"

"Visual Basic".Substring(0, 1) is "V"

IndexOf Method

Let *str1* and *str2* be strings.

```
str1.IndexOf(str2)
```

is the position of the first occurrence of *str2* in *str1*.

(**Note:** Has value -1 if *str2* is not a substring of *str1*.)

`"Visual Basic".IndexOf("is")` is 1.

`"Visual Basic".IndexOf("si")` is 9.

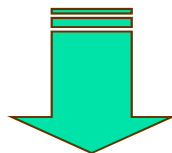`"Visual Basic".IndexOf("ab")` is -1.

The Empty String

- The string **""**, which contains no characters, is called the **empty string** or the **zero-length string.**

- The statement `lstBox.Items.Add("")` skips a line in the list box.

- The contents of a text box can be cleared with either the statement

  `txtBox.Clear()`

- or the statement

  `txtBox.Text = ""`

Trim method

- Removes all occurrences of white space characters from the beginning and end of this instance.
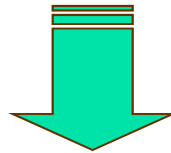
- E.g. "   VTC YC   ".trim

"VTC YC"

String property- Length

- Gets the number of characters in the string
- E.g. "Quick Test".length

10 *characters*

ToUpper & ToLower

- ToUpper: Converts all characters in a string to uppercase

- ToLower: Converts all characters in a string to lowercase

- E.g.
  "apple".toUpper ➔ "APPLE"
  "ApPlE".toLower ➔ "apple"

Initial Value of a String

- By default the initial value is **Nothing**
- Strings can be given a different initial value as follows:

```
Dim today As String = "Monday"
```

Option Strict

- Visual Basic allows numeric variables to be assigned strings and vice versa, a poor programming practice.

- To turn this feature off, put the following statement at the very top of the code window

`Option Strict On`

Option Strict On for All Programs

- Select Options from the Tools menu
- In left pane, expand Projects and Solution
- Select VB Defaults
- Set Option Strict to On

With Option Strict On

```
Dim dblVar As Double, intVar As Integer
Dim strVar As String
```

Not Valid:	Replace with:
`intVar = dblVar`	`intVar = CInt(dblVar)`
`dblVar = strVar`	`dblVar = CDbl(strVar)`
`strVar = intVar`	`strVar = CStr(intVar)`

Comments

```
Private Sub btnCompute_Click (...)
                       Handles btnCompute.Click
   'Calculate the balance in an account
   Dim rate As Double   'Annual rate of interest
   Dim curBalance As Double   'Current balance
```

Internal Documentation

1. Other people can easily understand the program.

2. You can understand the program when you read it later.

3. Long programs are easier to read because the purposes of individual pieces can be determined at a glance.

Automatic Colorization

Comments – green

String literals – maroon

Keywords – blue

Note: Keywords are words such as Sub, Handles, Private, With, and End that have special meaning in Visual Basic. They cannot be used as variable names.

Line-Continuation Character

- A long line of code can be continued on another line by using an underscore (_) preceded by a space

```
msg = "I'm going to make " & _
        "him an offer he can't refuse."
```

Comment on Example

Consider

```
lstBox.Items.Add(name & ": " & wage * hours)
```
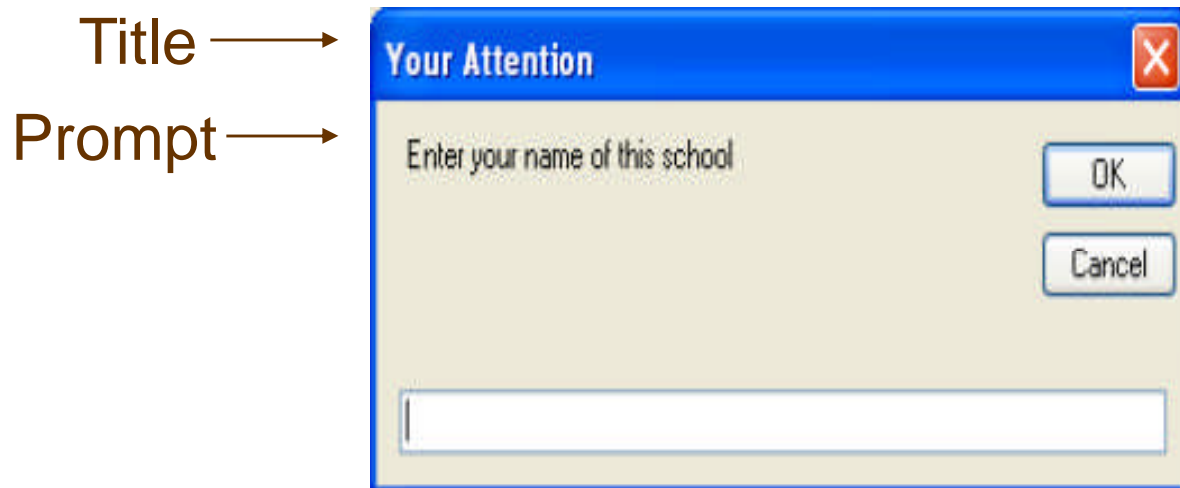
The ampersand automatically converted
`wage * hours` into a string before concatenating.
We didn't have to convert `wage * hours` with CStr.

Getting Input from an Input Dialog Box

```
Prompt = "Enter the name of this school"
stringVar = InputBox(prompt, "Your Attention")
SchoolName = InputBox("Enter the name " _
    & "of this school", "Your Attention")
```

Title ⟶

Prompt ⟶

Your Attention

Enter your name of this school

OK

Cancel

Using a Message Dialog Box for Output

```
MsgBox(prompt, 0, title)

MsgBox("Nice try, but no cigar.", 0, _
                        "Consolation")
```

Title ⟶ **Consolation**

Prompt ⟶ Nice try, but no cigar.

OK